

ADITI AKARSH

1BM19CS007

OOJ LAB

LAB PROGRAMS

LAB 1

Develop a Java program that prints all real solutions to the quadratic equation $ax^2 + bx + c = 0$. Read in a , b , c and use the quadratic formula. If the discriminant $b^2 - 4ac$ is negative, display a message stating that there are no real solutions.

```
import java.util.*;
class Main
{
    public static void main(String args[])
    {
        double r1,r2;

        Scanner scan = new Scanner(System.in);
        System.out.println("Enter the coefficients a, b, c: ");
        double a = scan.nextFloat();
        double b = scan.nextFloat();
        double c = scan.nextFloat();
        double d=b*b-4*a*c;
        if(d>0)
        {
            r1=(-b+Math.sqrt(d))/(2*a);
            r2=(-b-Math.sqrt(d))/(2*a);
            System.out.printf("root1=%.2f and root2=%.2f",r1,r2);
        }
        else if(d==0)
        {
            r1=r2=-b/(2*a);
            System.out.printf("root1=root2=%.2f",r1);
        }
        else
        {
            double r=-b/(2*a);
            double i=Math.sqrt(-d)/(2*a);
            System.out.printf("root1=%.2f+%.2fi and root2=%.2f-%.2fi",r,i,r,i);
        }
    }
}
```

OUTPUT:

The screenshot shows an online Java compiler interface. The code in the editor is:

```
1 import java.util.*;
2 class Main
3 {
4     public static void main(String args[])
5     {
6         double r1,r2;
7
8         Scanner scan = new Scanner(System.in);
9         System.out.print("Enter the coefficients a, b, c: ");
10        double a = scan.nextFloat();
11        double b = scan.nextFloat();
12        double c = scan.nextFloat();
13        double d=b*b-4*a*c;
14        if(d>0)
15        {
16            r1=(-b+Math.sqrt(d))/(2*a);
17            r2=(-b-Math.sqrt(d))/(2*a);
18            System.out.printf("root1=%2f and root2=%2f",r1,r2);
19        }
20        else if(d==0)
21        {
22            r1=r2=-b/(2*a);
23            System.out.printf("root1=root2=%2f",r1);
24        }
25        else
26        {
27            double r=-b/(2*a);
28            double i=Math.sqrt(-d)/(2*a);
29            System.out.printf("root1=%2f%2fi and root2=%2f-%2fi",r,i,r,i);
30        }
31    }
32 }
```

The terminal window shows the input "1 3 6" and the output "root1=-1.50+1.94i and root2=-1.50-1.94i".

...Program finished with exit code 0

The screenshot shows an online Java compiler interface. The code in the editor is identical to the one in the first screenshot:

```
1 import java.util.*;
2 class Main
3 {
4     public static void main(String args[])
5     {
6         double r1,r2;
7
8         Scanner scan = new Scanner(System.in);
9         System.out.print("Enter the coefficients a, b, c: ");
10        double a = scan.nextFloat();
11        double b = scan.nextFloat();
12        double c = scan.nextFloat();
13        double d=b*b-4*a*c;
14        if(d>0)
15        {
16            r1=(-b+Math.sqrt(d))/(2*a);
17            r2=(-b-Math.sqrt(d))/(2*a);
18            System.out.printf("root1=%2f and root2=%2f",r1,r2);
19        }
20        else if(d==0)
21        {
22            r1=r2=-b/(2*a);
23            System.out.printf("root1=root2=%2f",r1);
24        }
25        else
26        {
27            double r=-b/(2*a);
28            double i=Math.sqrt(-d)/(2*a);
29            System.out.printf("root1=%2f%2fi and root2=%2f-%2fi",r,i,r,i);
30        }
31    }
32 }
```

The terminal window shows the input "1 3 6" and the output "root1=-1.50+1.94i and root2=-1.50-1.94i".

...Program finished with exit code 0

WRITEUP:

Expt. No. Lab Program 1

Date _____

Page No. _____

```
# import java.util.*;
class Main
{
    public static void main (String args[])
    {
        double x1, x2;
        Scanner scan = new Scanner (System.in);
        System.out.println ("Enter the coefficients a, b, c : ");
        double a = scan.nextDouble ();
        double b = scan.nextDouble ();
        double c = scan.nextDouble ();
        double d = (b * b) - (4 * a * c);
        if (d > 0)
        {
            x1 = (-b + Math.sqrt (d)) / (2 * a);
            x2 = (-b - Math.sqrt (d)) / (2 * a);
            System.out.printf ("root 1 = %.2f and\n"
                "root 2 = %.2f", x1, x2);
        }
        else if (d == 0)
        {
            x1 = x2 = -b / (2 * a);
            System.out.printf ("root 1 = root 2 = %.2f", x1);
        }
        else
        {
            double r1 = Math.sqrt (-a) / (2 * a);
            double r2 = b / (2 * a);
            System.out.printf ("root 1 = %.2f + %.2f i\n"
                "and root 2 = %.2f - %.2f i", r1, r2, r1, r2);
        }
    }
}
```

Teacher's Signature _____

LAB 2

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

```
import java.util.*;
class Student {
    private String usn;
    private String name;
    private int n;
    private int credits[] = new int[n];
    private int marks[] = new int[n];

    void accept()
    {
        Scanner s=new Scanner(System.in);
        System.out.println("Enter student details");
        System.out.println("USN:");
        usn=s.nextLine();
        System.out.println("Name:");
        name=s.nextLine();
        System.out.println("Enter the number of subjects:");
        n=s.nextInt();
        System.out.println("Enter credits and marks attained by the student in each subject");
        for(int i=0;i<n;i++)
        {
            credits[i]=s.nextInt();
            marks[i]=s.nextInt();
        }
    }
    void display()
    {
        System.out.println("Student details:");
        System.out.println("USN:"+usn);
        System.out.println("Name:"+name);
        System.out.println("Marks in each subject:");
        for(int i=0;i<n;i++)
        {
            System.out.println("Subject "+(i+1)+":"+marks[i]);
        }
    }
    double calc()
    {
        int tcp=0,tc=0;
```

```
for(int i=0;i<n;i++)
{
    tc=tc+credits[i];
    if(marks[i]>=40)
    {
        tcp=tcp+(((marks[i]/10)+1)*credits[i]);
    }
}
return (double)tcp/tc;
}
}

class Main
{
public static void main(String args[])
{
    Student s1=new Student();
    s1.accept();
    s1.display();
    System.out.println("SGPA: "+s1.calc());
}
}
```

OUTPUT:

C:\Users\Adith\Desktop\StudentJava - Notepad++

File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?

labprog1.java edaprogram2.java evenodd.java indicesarray.java Student.java

```
1 import java.util.*;
2 class Student
3 {
4     private String USN;
5     private String name;
6     private int n;
7     private double SGPA = 0;
8     private int totalCredits = 0;
9     private int credits[][];
10    private double marks[][];
11    Scanner in = new Scanner(System.in);
12
13    void accept()
14    {
15        System.out.println("Enter USN of the student");
16        USN = in.nextLine();
17        System.out.println("Enter Name of the student");
18        name = in.nextLine();
19        System.out.println("Enter number of subjects");
20        n = in.nextInt();
21        credits = new int[n];
22        marks = new double[n];
23        System.out.println("Details of the subjects:");
24        for(int i=0;i<n;i++)
25        {
26            System.out.println("Enter credits for subject "+(i+1));
27            credits[i] = in.nextInt();
28            System.out.println("Enter marks for subject "+(i+1));
29            marks[i] = in.nextInt();
30            Calculate(credits[i],marks[i],i);
31        }
32    }
33
34
35    void Calculate(int credit,double mark,int j)
36    {
37        totalCredits = totalCredits + credit;
38        if(mark==100 && mark<=100)
39            SGPA = SGPA + ((1*credit));
40        else if(mark>=80 && mark<=89)
41            SGPA = SGPA + ((1*credit));
42        else if(mark>=70 && mark<=79)
43            SGPA = SGPA + ((1*credit));
44        else if(mark>=60 && mark<=69)
```

C:\Users\Adit\Desktop\Student.java - Notepad++

```
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
abprog.java exaprog.java evenodd.java indicesarray.java Student.java

27 credits[i] = in.nextInt();
28 System.out.println("Enter marks for subject "+(i+1));
29 marks[i] = in.nextInt();
30 Calculate(credits[i],marks[i],i);
31 }
32 }
33
34 void Calculate(int credit,double mark,int j)
35 {
36     totalCredits = totalCredits + credit;
37     if(mark>=90 &&mark<=100)
38         SGPA = SGPA + ((10*credit));
39     else if(mark>=80 &&mark<=90)
40         SGPA = SGPA + ((9*credit));
41     else if(mark>=70 &&mark<=80)
42         SGPA = SGPA + ((8*credit));
43     else if(mark>=60 &&mark<=70)
44         SGPA = SGPA + ((7*credit));
45     else if(mark>=50 &&mark<=60)
46         SGPA = SGPA + ((6*credit));
47     else if(mark>=40 &&mark<=50)
48         SGPA = SGPA + ((5*credit));
49     else if(mark>=0 &&mark<=40)
50         SGPA = SGPA + ((4*credit));
51     else
52         System.out.println("Failed in Subject "+(j+1));
53 }
54
55 void Display()
56 {
57     System.out.println("Details of the Student");
58     System.out.println("USN: "+USN);
59     System.out.println("Name :"+name);
60     System.out.println("SGPA of Student "+(SGPA/totalCredits));
61 }
62 }
63 }
64 class Main {
65 public static void main(String args[])
66 {
67     Student s1 = new Student();
68     s1.accept();
69     s1.Display();
70 }
```

Java source file

length:2001 lines:70 Ln:57 Col:54 Sel:0|0 Windows (CR LF) UTF-8 INS

Online Java Compiler - online ed | aditiakarsh/eojlab | New Tab | +

input

```
Enter USN of the student
lmbhjk23
Enter Name of the student
SREEJA
Enter number of subjects
5
Details of the subjects:
Enter credits for subject 1
2
Enter marks for subject 1
80
Enter credits for subject 2
4
Enter marks for subject 2
90
Enter credits for subject 3
4
Enter marks for subject 3
90
Enter credits for subject 4
2
Enter marks for subject 4
89
Enter credits for subject 5
2
Enter marks for subject 5
78
Details of the Student
USN: lmbhjk23
Name :SREEJA
SGPA of Student 9.428571428571429

...Program finished with exit code 0
Press ENTER to exit console.
```

Type here to search

21:01 06-10-2020

WRITEUP:

IBM19CS007

Expt. No. Lab Program 2

Date 6/10/2020

Page No. _____

```
import java.util.*;
class Student
{
    private String USN;
    private String name;
    private int n;
    private double SGPA = 0;
    private int totalCredits = 0;
    private int credits[];
    private double marks[];
    Scanner in = new Scanner(System.in);
    void accept()
    {
        System.out.println("Enter USN");
        USN = in.nextLine();
        System.out.println("Enter Name");
        name = in.nextLine();
        System.out.println("Enter number of subjects");
        n = in.nextInt();
        credits = new int[n];
        marks = new double[n];
        System.out.println("Details of the subjects:");
        for (int i = 0; i < n; i++)
        {
            System.out.println("Enter credits" + (i + 1));
            credits[i] = in.nextInt();
            calculate(credits[i], marks[i], i);
        }
    }
}
```

Teacher's Signature _____

IBM19C8007

Date 6/10/2020

Expt. No. Lab Program 2

```

void calculate() int credit, double mark, int j)
{
    total credits = totalCredits + credit;
    if (mark >= 90 && mark <= 100)
        SGPA = SGPA + (10 * credit);
    else if (mark >= 80 && mark <= 89)
        SGPA = SGPA + (9 * credit);
    else if (mark >= 70 && mark <= 79)
        SGPA = SGPA + (8 * credit);
    else if (mark >= 40 && mark <= 49)
        SGPA = SGPA + (5 * credit);
    else
        System.out.println("Failed");
}

void Display()
{
    System.out.println("Details:");
    System.out.println("USN: " + USN);
    System.out.println("Name: " + Name);
    System.out.println("SGPA: " + (SGPA / totalCredits));
}

```

```
3
class Main {
    public static void main (String args[]) {
        Student s1 = new Student ();
        s1.accept ();
        s1.Display ();
    }
}
```

Teacher's Signature

LAB 3

Create a class Book which contains four members: name, author, price,num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a `toString()` method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.*;  
class Book {  
    String name;  
    String author;  
    float price;  
    int num_pages;  
    Book()  
    {}  
    Book(String name,String author,int price,int num_pages)  
    {  
        this.name=name;  
        this.author=author;  
        this.price=price;  
        this.num_pages=num_pages;  
    }  
    void display()  
    {  
        Scanner inp=new Scanner(System.in);  
        System.out.println("Enter name of book:");  
        name=inp.next();  
        System.out.println("Enter author of book:");
```

```

author=inp.next();
System.out.println("Enter price of book:");
price=inp.nextFloat();
System.out.println("Enter number of pages of book:");
num_pages=inp.nextInt();

}

public String toString()
{
    return ("Name: "+name + "\n" + "Author: "+author + "\n" + "Price: "+price + "\n"
+"Number of pages: "+num_pages );
}

}

class BookMain {
    public static void main(String args[])
    {
        Scanner a=new Scanner(System.in);
        System.out.println("Enter the number of books:");
        int n=a.nextInt();
        Book b[]=new Book[n];
        for(int i=0;i<n;i++)
        {
            b[i]=new Book();
            System.out.println("Enter the details of "+(i+1)+" book");
            b[i].display();
        }
        for(int i=0;i<n;i++)
        {
            System.out.println("Details of book "+(i+1));
            System.out.println(b[i]);
        }
    }
}

```

OUTPUT:

```
1 import java.util.*;
2 class Book {
3     String name;
4     String author;
5     float price;
6     int num_pages;
7 }
8
9 Book(String name,String author,int price,int num_pages)
10 {
11     this.name=name;
12     this.author=author;
13     this.price=price;
14     this.num_pages=num_pages;
15 }
16
17 void display()
18 {
19     Scanner inp=new Scanner(System.in);
20     System.out.println("Enter name of book:");
21     name=inp.nextLine();
22     System.out.println("Enter author of book:");
23     author=inp.nextLine();
24     System.out.println("Enter price of book:");
25     price=inp.nextFloat();
26     System.out.println("Enter number of pages of book:");
27     num_pages=inp.nextInt();
28 }
29
30 public String toString()
31 {
32     return ("Name: "+name + "\n" + "Author: "+author + "\n" + "Price: "+price + "\n" +"Number of pages: "+num_pages );
33 }
34
35 class BookMain {
36     public static void main(String args[])
37     {
38         Scanner a=new Scanner(System.in);
39         System.out.println("Enter the number of books:");
40         int n=a.nextInt();
41         Book b[] =new Book[n];
42         for(int i=0;i<n;i++)
43         {
44             b[i]=new Book();
45             System.out.println("Enter the details of "+(i+1)+" book");
46             b[i].display();
47         }
48         for(int i=0;i<n;i++)
49         {
50             System.out.println("Details of book "+(i+1));
51             System.out.println(b[i]);
52         }
53     }
54 }
```

Java source file

Type here to search

length: 1,242 lines: 52 Ln: 13 Col: 18 Sel: 0 | 0 Windows (CR LF) UTF-8 INS

1543 13-10-2020

```
1 import java.util.*;
2 class Book {
3     String name;
4     String author;
5     float price;
6     int num_pages;
7 }
8
9 Book(String name,String author,int price,int num_pages)
10 {
11     this.name=name;
12     this.author=author;
13     this.price=price;
14     this.num_pages=num_pages;
15 }
16
17 void display()
18 {
19     Scanner inp=new Scanner(System.in);
20     System.out.println("Enter name of book:");
21     name=inp.nextLine();
22     System.out.println("Enter author of book:");
23     author=inp.nextLine();
24     System.out.println("Enter price of book:");
25     price=inp.nextFloat();
26     System.out.println("Enter number of pages of book:");
27     num_pages=inp.nextInt();
28 }
29
30 public String toString()
31 {
32     return ("Name: "+name + "\n" + "Author: "+author + "\n" + "Price: "+price + "\n" +"Number of pages: "+num_pages );
33 }
34
35 class BookMain {
36     public static void main(String args[])
37     {
38         Scanner a=new Scanner(System.in);
39         System.out.println("Enter the number of books:");
40         int n=a.nextInt();
41         Book b[] =new Book[n];
42         for(int i=0;i<n;i++)
43         {
44             b[i]=new Book();
45             System.out.println("Enter the details of "+(i+1)+" book");
46             b[i].display();
47         }
48         for(int i=0;i<n;i++)
49         {
50             System.out.println("Details of book "+(i+1));
51             System.out.println(b[i]);
52         }
53     }
54 }
```

Java source file

Type here to search

length: 1,242 lines: 52 Ln: 13 Col: 18 Sel: 0 | 0 Windows (CR LF) UTF-8 INS

1543 13-10-2020

```
Command Prompt
Microsoft Windows [Version 10.0.18362.1082]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Aditi>cd C:\Users\Aditi\Desktop
C:\Users\Aditi\Desktop>javac Book.java
C:\Users\Aditi\Desktop>java BookMain
Enter the number of books:
1
Enter the details of 1 book
Enter name of book:
red
Enter author of book:
tre
Enter price of book:
334
Enter number of pages of book:
76
Details of book 1
Name: red
Author: tre
Price: 334.0
Number of pages: 76
C:\Users\Aditi\Desktop>
```

WRITEUP:

Expt. No. Lab Program 3

Date _____

Page No. _____

```
import java.util.*;
class Book
{
    String name;
    String author;
    float price;
    int num_pages;
    Book()
    {
        this.name = name;
        this.author = author;
        this.price = price;
        this.num_pages = num_pages;
    }
    void display()
    {
        Scanner inp = new Scanner(System.in);
        System.out.println("Enter the name of the book");
        name = inp.nextLine();
        System.out.println("Enter the author of book");
        author = inp.nextLine();
        System.out.println("Enter the price");
        price = inp.nextFloat();
        System.out.println("Enter the pages");
    }
}
```

Teacher's Signature _____

Expt. No. Lab program 3

Date _____

Page No. _____

```
public String toString()
{
    return "Name:" + name + "\n" + "Author:" + author
           + "\n" + "Price:" + price + "\n" + "Number of
           pages" + numPages;
}
```

```
class BookMain {
}
```

```
public static void main (String args[])
{
}
```

```
Scanner a = new Scanner (System. in);
System.out.println ("Enter the number of books");
int n = a.nextInt ();
Book b[] = new Book [n];
for (int i = 0; i < n; i++)
{
}
```

```
    b[i] = new Book ();
    System.out.println ("Enter the details of"
                       + (i + 1) + " book");
    b[i].display ();
}
```

```
for (int i = 0; i < n; i++)
{
}
```

```
    System.out.println ("Details of book"
                       + (i + 1));
    System.out.println (b[i]);
}
}
```

Teacher's Signature _____

LAB 4

Develop a Java program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contain only the method printArea() that prints the area of the given shape.

```
abstract class Shape {
    double dim1, dim2;
    abstract double printArea();
}
```

```
class Rectangle extends Shape {
    Rectangle(double a, double b) {
        dim1 = a;
        dim2 = b;
    }
}
```

```

}

double printArea(){
    System.out.println("Inside the Rectangle");
    return dim1*dim2;
}
}

class Triangle extends Shape{
    Triangle(double a, double b){
        dim1 = a;
        dim2 = b;
    }
    double printArea(){
        System.out.println("Inside the Triangle");
        return dim1*dim2/2;
    }
}

class Circle extends Shape{
    Circle(double a){
        dim1 = a;
    }
    double printArea(){
        System.out.println("Inside the Circle");
        return 3.14*dim1*dim1;
    }
}

class abs1Main{
    public static void main(String args[]){
        Rectangle r = new Rectangle(10,20);
        Triangle t = new Triangle(20,30);
        Circle c = new Circle(35);

        System.out.println("Area of Rectangle is:" +r.printArea());
        System.out.println("Area of Triangle is:" +t.printArea());
        System.out.println("Area of Circle is:" +c.printArea());
    }
}

```

OUTPUT:

The screenshot shows a Windows desktop environment. In the center is a Microsoft Edge browser window titled "Courses" with the URL "onlinegdb.com/online_java_compiler". The browser interface includes tabs for "Courses", "ojlab/WEEK8 at master · aditiak ·", and "Online Java Compiler - online ed...". Below the tabs are standard browser controls: back, forward, search, and refresh. The main content area of the browser displays Java code in a dark-themed code editor. The code defines three classes: Shape, Rectangle, and Triangle, each with a printArea() method that prints its area and a message indicating it's inside itself. The output window shows the execution results: Inside the Rectangle, Area of Rectangle is:200.0; Inside the Triangle, Area of Triangle is:300.0; Inside the Circle, Area of Circle is:3846.5. At the bottom of the browser window, there is a message "...Program finished with exit code 0 Press ENTER to exit console." The browser is positioned above a taskbar which contains several pinned PDF files labeled "1BM19CS007_AdI...pdf" and a search bar. The system tray in the bottom right corner shows the date as 04-11-2020 and the time as 00:17.

```
Main.java
1- abstract class Shape{
2 double dim1,dim2;
3 abstract double printArea();
4 }
5
6 class Rectangle extends Shape{
7 Rectangle(double a, double b){
8 dim1 = a;
9 dim2 = b;
10 }
11 double printArea(){
12 System.out.println("Inside the Rectangle");
13 return dim1*dim2;
14 }
15 }
16
17 class Triangle extends Shape{
18 Triangle(double a, double b){
19 dim1 = a;
20 dim2 = b;
21 }
22 double printArea(){
23 System.out.println("Inside the Triangle");
24 return dim1*dim2/2;
25 }
26 }
27
28 class Circle extends Shape{
29 Circle(double a){
30 dim1 = a;
31 }
32 double printArea(){
33 System.out.println("Inside the Circle");
34 return 3.14*dim1*dim1;
35 }
36 }
37
38 class Main{
39 public static void main(String args[]){
40 Rectangle r = new Rectangle(10,20);
41 Triangle t = new Triangle(20,30);
42 Circle c = new Circle(35);
43
44 System.out.println("Area of Rectangle is:" +r.printArea());
45 System.out.println("Area of Triangle is:" +t.printArea());
46 System.out.println("Area of Circle is:" +c.printArea());
47 }
48 }
```

This screenshot is nearly identical to the one above, showing the same browser window with Java code and its output. The code defines the same three classes: Shape, Rectangle, and Triangle, with their respective printArea() methods and area calculations. The output shows the areas of each shape: Inside the Rectangle, Area of Rectangle is:200.0; Inside the Triangle, Area of Triangle is:300.0; Inside the Circle, Area of Circle is:3846.5. The browser is again positioned above a taskbar with multiple pinned PDF files and a search bar, and the system tray shows the date as 04-11-2020 and the time as 00:17.

```
Main.java
19 dim1 = a;
20 dim2 = b;
21 }
22 double printArea(){
23 System.out.println("Inside the Triangle");
24 return dim1*dim2/2;
25 }
26 }
27
28 class Circle extends Shape{
29 Circle(double a){
30 dim1 = a;
31 }
32 double printArea(){
33 System.out.println("Inside the Circle");
34 return 3.14*dim1*dim1;
35 }
36 }
37
38 class Main{
39 public static void main(String args[]){
40 Rectangle r = new Rectangle(10,20);
41 Triangle t = new Triangle(20,30);
42 Circle c = new Circle(35);
43
44 System.out.println("Area of Rectangle is:" +r.printArea());
45 System.out.println("Area of Triangle is:" +t.printArea());
46 System.out.println("Area of Circle is:" +c.printArea());
47 }
48 }
```

WRITEUP:

Expt. No. Lab Program 4

Date _____

Page No. _____

abstract class Shape {

 double dim1, dim2;

 abstract double printArea();

}

class Rectangle extends Shape {

 Rectangle (double a, double b)

{

 dim1 = a;

 dim2 = b;

}

 double printArea()

{

 System.out.println ("Inside the Rectangle");

 return dim1 * dim2;

}

}

class Triangle extends Shape {

 Triangle (double a, double b) {

 dim1 = a;

 dim2 = b;

}

 double printArea()

 System.out.println ("Inside the Triangle");

 return dim1 * dim2;

}

}

Teacher's Signature _____

Class Circle extends Shape {

Circle (double) {

dim1 = a;

y
double pntArea();

System.out.println("Inside the circle");

when $\pi \cdot 14 \cdot dim1^2 < dim2$;

}

Class abs1Main

public static void main (String args[]) {

Rectangle r = new Rectangle(10, 20);

Triangle t = new Triangle(20, 30);

Circle c = new Circle(35);

System.out.println("Area of Rectangle is" +

r.pntArea());

System.out.println("Area of Triangle is" +

t.pntArea());

System.out.println("Area of Circle is" +

c.pntArea());

}

Teacher's Signature _____

LAB 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of

account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks:

- Accept deposit from customer and update the balance.
- Display the balance.
- Compute and deposit interest
- Permit withdrawal and update the balance
- Check for the minimum balance, impose penalty if necessary and update the balance.

```
Importjava.util.
```

```
Scanner;
```

```
abstract class Account{
    String c_name, acc_type;
    int acc_num;
    double balance;
    int minbalance = 2000;
    Account(String c_name, int acc_num, double balance){
        this.c_name = c_name;
        this.acc_num = acc_num;
        this.balance = balance;
        this.acc_type = acc_type;
    }

    abstract void addbal(double amount);
    abstract void display();
    abstract void withdraw(double amount);
}

class curr_acct extends Account{
    curr_acct(String c_name, int acc_num, double balance){
        super(c_name, acc_num, balance);
        System.out.println("Details of the customer:");
        System.out.println("Customer name: " +c_name +"\tAccount number: " +acc_num+ "\tBalance: " +balance+ "Account type: current");
    }

    void addbal(double amount){
        this.balance += amount;
    }

    void display(){
        System.out.println("The balance is:" +this.balance);
    }

    void withdraw(double amount){
        if(this.balance<amount){
            System.out.println("Insufficient funds");
            System.out.println("Your balance is: " +this.balance);
            return;
        }
        this.balance = this.balance - amount;
    }
}
```

```

        if(this.balance<minbalance){
            this.balance = this.balance - this.balance*0.5;
            System.out.println("A penalty of Rs. "
+this.balance*0.5 + "has been charged as minimum balance is
not satisfied");
            System.out.println("Updated Balance: "
+this.balance);
            System.out.println("Cannot withdraw");
        }
        else if(balance> minbalance){
            this.balance = this.balance - amount;
            System.out.println("Balance is: " +this.balance);
        }
    }
}

class sav_acct extends Account{
    sav_acct(String c_name, int acc_num, double balance){
        super(c_name, acc_num, balance);
        System.out.println("Customer name: " +c_name +"\tAccount
number: " +acc_num+ "\tBalance: " +balance+ "Account type:
savings");
    }
    void addbal(double amount){
        this.balance += amount;
    }

    void display(){
        System.out.println("The balance is:" +this.balance);
    }

    void withdraw(double amount){
        if(this.balance<amount){
            System.out.println("Insufficient funds");
            System.out.println("Yor balance is: " +this.balance);
        }
        this.balance = this.balance - amount;
        if(this.balance<minbalance){
            this.balance = this.balance - this.balance*0.5;
            System.out.println("A penalty of Rs. "
+this.balance*0.5 + "has been charged as minimum balance is
not satisfied");
            System.out.println("Updated Balance: "
+this.balance);
            System.out.println("Cannot withdraw");
        }
        else if(balance> minbalance){
            this.balance = this.balance - amount;
            System.out.println("Balance is: " +this.balance);
        }
    }
}

```

```

void intrest(double amount){
    int time = 3, n=1;
    System.out.println("Rate of intrest is 0.2");
    this.balance = this.balance*Math.pow(1+(0.2)/n,
(n*time));
}
}

class abs2Main{
public static void main(String args[]){
    int choice,ch,n=1;
    double amount;
    Scanner s1 = new Scanner(System.in);
    curr_acct c = new curr_acct("Aditi", 12345, 50000);
    sav_acct s = new sav_acct("Aditi", 12345, 50000);
    System.out.println("Press 1.For Current
account\nPress 2.For Savings account");
    choice = s1.nextInt();
    switch(choice){
        case 1: System.out.println("****Current
Account****");
            while(n!=0){
                System.out.println("1.AddBalance\n2.displayBalance\n3.withd
raw\n4.checkbook\n5.Exit");
                ch = s1.nextInt();
                String reciever;
                double recamount;
                switch(ch){
                    case 1:
                        System.out.println("enter
amount to be added:");
                        amount = s1.nextDouble();
                        c.addbal(amount);
                        break;

                    case 2:
                        c.display();
                        break;

                    case 3:
                        System.out.println("enter
amount to be withdrawn:");
                        amount = s1.nextDouble();
                        c.withdraw(amount);
                        break;

                    case 4:
                        System.out.println("Enter
the name of the reciever:");
                }
            }
        }
    }
}

```



```

        amount = s1.nextDouble();
        s.withdraw(amount);
        break;

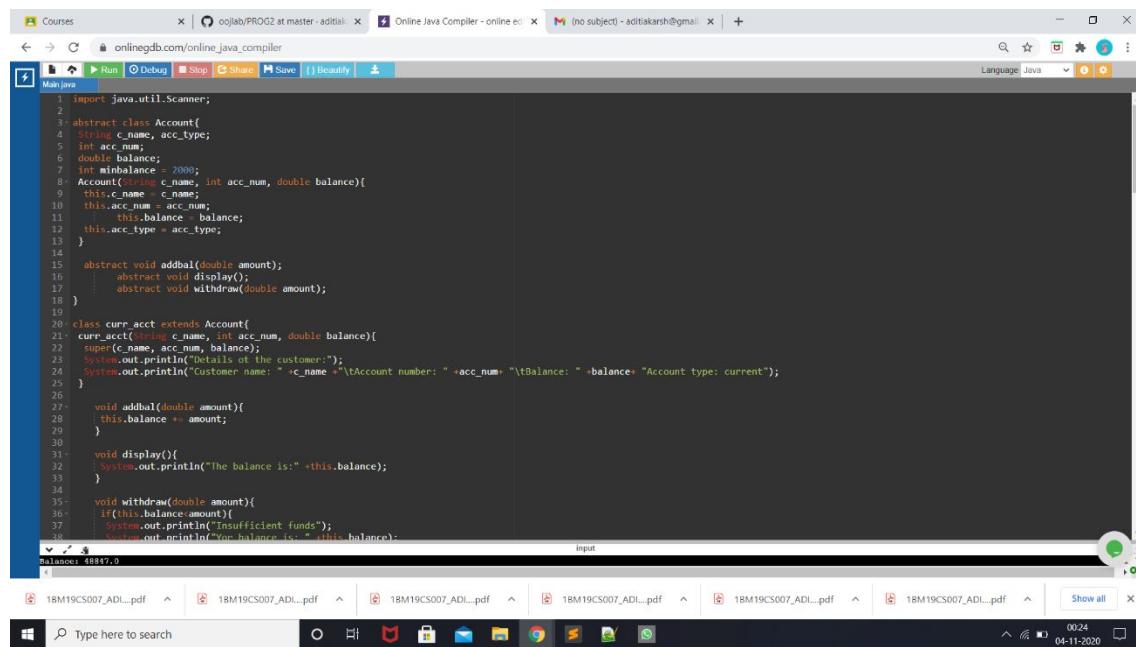
    case 4:
        n=0;

    default:
        System.out.println("Invalid input");
    }
}

default: System.out.println("Invalid input");
}
}
}

```

OUTPUT:

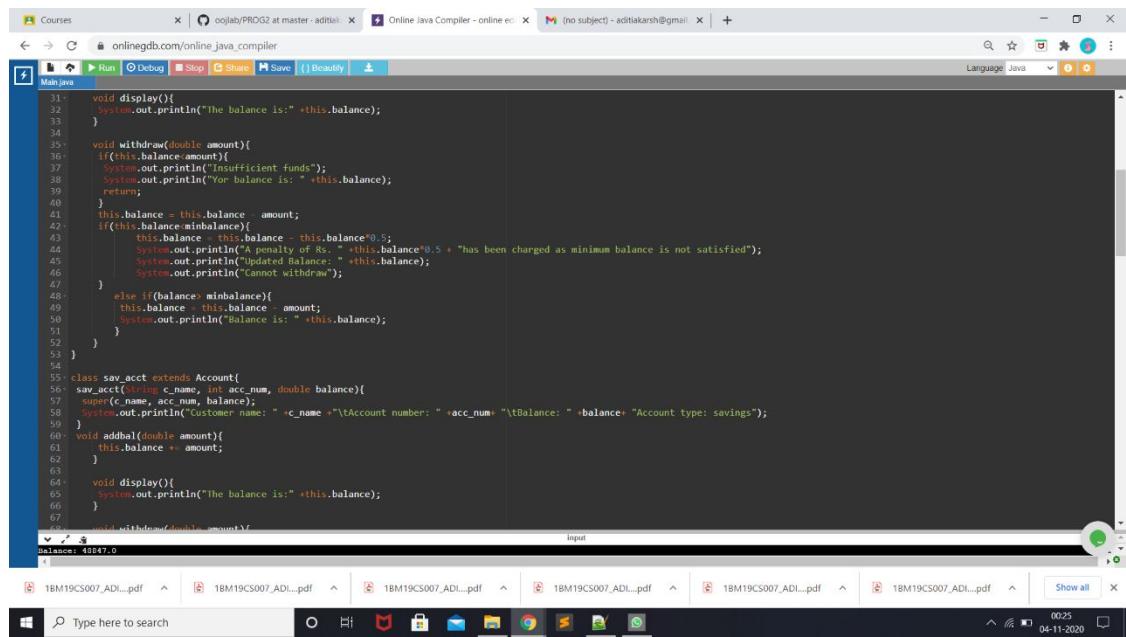


```

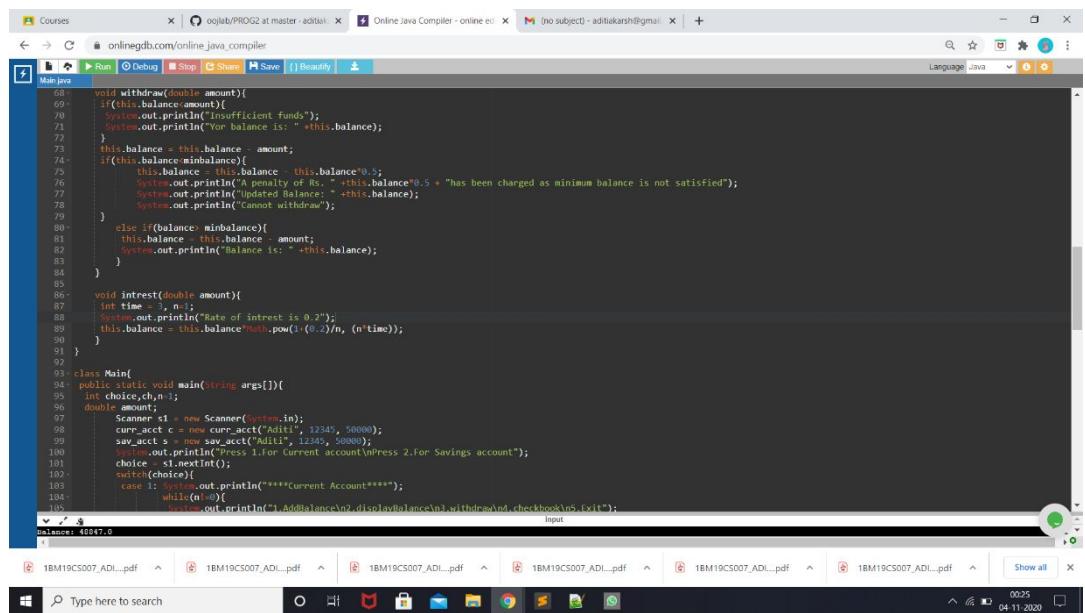
1 import java.util.Scanner;
2
3 abstract class Account{
4     String c_name, acc_type;
5     int acc_num;
6     double balance;
7     int minbalance = 2000;
8     Account(String c_name, int acc_num, double balance){
9         this.c_name = c_name;
10        this.acc_num = acc_num;
11        this.balance = balance;
12        this.acc_type = acc_type;
13    }
14
15    abstract void addbal(double amount);
16    abstract void display();
17    abstract void withdraw(double amount);
18 }
19
20 class curr_acct extends Account{
21     curr_acct(String c_name, int acc_num, double balance){
22         super(c_name, acc_num, balance);
23         System.out.println("Details of the customer:");
24         System.out.println("Customer name: " +c_name +"\tAccount number: " +acc_num+ "\tBalance: " +balance+ "Account type: current");
25     }
26
27     void addbal(double amount){
28         this.balance += amount;
29     }
30
31     void display(){
32         System.out.println("The balance is: " +this.balance);
33     }
34
35     void withdraw(double amount){
36         if(this.balance > amount)
37             System.out.println("Insufficient funds");
38         else
39             System.out.println("Your balance is: " +this.balance);
40     }
41 }
42
43 public class Main {
44     public static void main(String[] args) {
45         Scanner s = new Scanner(System.in);
46         curr_acct s1 = new curr_acct("Aditi", 1234567890, 40000.0);
47         System.out.println("Enter the amount to withdraw: ");
48         double amount = s.nextDouble();
49         s1.withdraw(amount);
50         System.out.println("Your balance is: " +s1.balance);
51     }
52 }

```

The output window shows the final balance: Balance: 40000.0



```
31+ void display(){
32     System.out.println("The balance is:" +this.balance);
33 }
34
35+ void withdraw(double amount){
36     if(this.balance<amount)
37         System.out.println("Insufficient funds");
38     System.out.println("Your balance is:" +this.balance);
39     return;
40 }
41 this.balance = this.balance - amount;
42 if(this.balance<minBalance){
43     this.balance = this.balance - this.balance*0.5;
44     System.out.println("A penalty of Rs. " +this.balance*0.5 + " has been charged as minimum balance is not satisfied");
45     System.out.println("Updated Balance: " +this.balance);
46     System.out.println("Cannot withdraw");
47 }
48 else if(balance<minBalance){
49     this.balance = this.balance - amount;
50     System.out.println("Balance is: " +this.balance);
51 }
52 }
53 }
54
55 class sav_act extends Account{
56     String c_name, acc_num, balance;
57     sav_act(String name, acc_num, balance){
58         System.out.println("Customer name: " +c_name +"\tAccount number: " +acc_num +"\tBalance: " +balance+ "Account type: savings");
59     }
60     void addbal(double amount){
61         this.balance += amount;
62     }
63
64     void display(){
65         System.out.println("The balance is:" +this.balance);
66     }
67
68+ void withdraw(double amount){
69     if(this.balance<amount)
70         System.out.println("Insufficient funds");
71     System.out.println("Your balance is:" +this.balance);
72 }
73 this.balance = this.balance - amount;
74 if(this.balance<minBalance){
75     this.balance = this.balance - this.balance*0.5;
76     System.out.println("A penalty of Rs. " +this.balance*0.5 + " has been charged as minimum balance is not satisfied");
77     System.out.println("Updated Balance: " +this.balance);
78     System.out.println("Cannot withdraw");
79 }
80 else if(balance<minBalance){
81     this.balance = this.balance - amount;
82     System.out.println("Balance is: " +this.balance);
83 }
84 }
85
86+ void intrest(double amount){
87     int time = 3, n;
88     System.out.println("Rate of interest is 0.2");
89     this.balance = this.balance*Math.pow((1+(0.2)/n, (n*time)));
90 }
91 }
92
93 class Main{
94     public static void main(String args[]){
95         Scanner s1 = new Scanner(System.in);
96         double amount;
97         Scanner s1 = new Scanner(System.in);
98         curr_act c = new curr_act("Aditi", 12345, 50000);
99         sav_act s = new sav_act("Aditi", 12345, 50000);
100        System.out.println("Press 1.For Current account\nPress 2.For Savings account");
101        choice = s1.nextInt();
102        switch(choice){
103            case 1: System.out.println("****Current Account****");
104            while(n>0){
105                System.out.println("1.AddBalance\n2.DisplayBalance\n3.Withdraw\n4.Checkbook\n5.Exit");
106                choice = s1.nextInt();
107                switch(choice){
108                    case 1: addBalance();
109                    case 2: displayBalance();
110                    case 3: withdraw();
111                    case 4: checkbook();
112                    case 5: System.out.println("Exiting");
113                }
114            }
115        }
116    }
117 }
```



```
31+ void display(){
32     System.out.println("The balance is:" +this.balance);
33 }
34
35+ void withdraw(double amount){
36     if(this.balance<amount)
37         System.out.println("Insufficient funds");
38     System.out.println("Your balance is:" +this.balance);
39     return;
40 }
41 this.balance = this.balance - amount;
42 if(this.balance<minBalance){
43     this.balance = this.balance - this.balance*0.5;
44     System.out.println("A penalty of Rs. " +this.balance*0.5 + " has been charged as minimum balance is not satisfied");
45     System.out.println("Updated Balance: " +this.balance);
46     System.out.println("Cannot withdraw");
47 }
48 else if(balance<minBalance){
49     this.balance = this.balance - amount;
50     System.out.println("Balance is: " +this.balance);
51 }
52 }
53 }
54
55 class sav_act extends Account{
56     String c_name, acc_num, balance;
57     sav_act(String name, acc_num, balance){
58         System.out.println("Customer name: " +c_name +"\tAccount number: " +acc_num +"\tBalance: " +balance+ "Account type: savings");
59     }
60     void addbal(double amount){
61         this.balance += amount;
62     }
63
64     void display(){
65         System.out.println("The balance is:" +this.balance);
66     }
67
68+ void withdraw(double amount){
69     if(this.balance<amount)
70         System.out.println("Insufficient funds");
71     System.out.println("Your balance is:" +this.balance);
72 }
73 this.balance = this.balance - amount;
74 if(this.balance<minBalance){
75     this.balance = this.balance - this.balance*0.5;
76     System.out.println("A penalty of Rs. " +this.balance*0.5 + " has been charged as minimum balance is not satisfied");
77     System.out.println("Updated Balance: " +this.balance);
78     System.out.println("Cannot withdraw");
79 }
80 else if(balance<minBalance){
81     this.balance = this.balance - amount;
82     System.out.println("Balance is: " +this.balance);
83 }
84 }
85
86+ void intrest(double amount){
87     int time = 3, n;
88     System.out.println("Rate of interest is 0.2");
89     this.balance = this.balance*Math.pow((1+(0.2)/n, (n*time)));
90 }
91 }
92
93 class Main{
94     public static void main(String args[]){
95         Scanner s1 = new Scanner(System.in);
96         curr_act c = new curr_act("Aditi", 12345, 50000);
97         sav_act s = new sav_act("Aditi", 12345, 50000);
98         System.out.println("Press 1.For Current account\nPress 2.For Savings account");
99         choice = s1.nextInt();
100        switch(choice){
101            case 1: System.out.println("****Current Account****");
102            while(n>0){
103                System.out.println("1.AddBalance\n2.DisplayBalance\n3.Withdraw\n4.Checkbook\n5.Exit");
104                choice = s1.nextInt();
105                switch(choice){
106                    case 1: addBalance();
107                    case 2: displayBalance();
108                    case 3: withdraw();
109                    case 4: checkbook();
110                    case 5: System.out.println("Exiting");
111                }
112            }
113        }
114    }
115 }
```

The screenshot shows a Java development environment with a code editor and a terminal window.

Code Editor (Main.java):

```
111         case 1:
112             System.out.println("enter amount to be added:");
113             amount = s1.nextDouble();
114             c.addbal(amount);
115             break;
116
117         case 2:
118             c.display();
119             break;
120
121         case 3:
122             System.out.println("enter amount to be withdrawn:");
123             amount = s1.nextDouble();
124             c.withdraw(amount);
125             break;
126
127         case 4:
128             System.out.println("Enter the name of the receiver:");
129             receiver = s1.next();
130             System.out.println("Enter amount to be debited to receiver:");
131             recamount = s1.nextDouble();
132             if(recamount < c.balance){
133                 System.out.println("Insufficient Balance");
134             }
135             else{
136                 System.out.println("Amount of " + recamount + " sent to " + receiver);
137                 c.balance = c.balance - recamount;
138                 System.out.println("Balance: " + c.balance);
139                 s.balance = c.balance;
140             }
141
142         case 5:
143             n=0;
144             break;
145
146     default: System.out.println("Invalid input");
147 }
148 }
```

Terminal Window:

```
input
```

The screenshot shows a Java development environment with a code editor and a terminal window.

Code Editor (Main.java):

```
140
141         case 5:
142             n=0;
143             break;
144
145     default: System.out.println("Invalid input");
146 }
147 }
148
149
150 case 2: System.out.println("****Savings Account****");
151 while(n!=0){
152     System.out.println("1.AddBalance\n2.displayBalance\n3.withdraw\n4.Exit");
153     ch = s1.next();
154     switch(ch){
155         case '1':
156             System.out.println("enter amount to be added:");
157             amount = s1.nextDouble();
158             s.addbal(amount);
159             break;
160
161         case '2':
162             s.display();
163             break;
164
165         case '3':
166             System.out.println("enter amount to be withdrawn:");
167             amount = s1.nextDouble();
168             s.withdraw(amount);
169             break;
170
171         case '4':
172             n=0;
173             break;
174
175     default: System.out.println("Invalid input");
176 }
177 }
```

Terminal Window:

```
https://www.onlinegdb.com/online_java_compiler#tab-stdin
```

The screenshot shows a Java IDE interface with multiple tabs open. The active tab displays a Java code snippet for a banking application. The code includes methods for adding balance, displaying balance, withdrawing, and checking account details. It also handles a minimum balance constraint and a transfer operation.

```
75     this.balance = 1815;balance = this.balance*0.5;
76     System.out.println("A penalty of Rs. - "+this.balance+" has been charged as minimum balance is not satisfied");
77 }
78
79 Details of the customer:
Customer name: Aditi Account number: 12345 Balance: 50000.0 account type: current
Customer name: Aditi Account number: 12345 Balance: 50000.0 account type: savings
Press 1.For Current account
Press 2.For Savings account
****Current Account****
1.addBalance
2.displayBalance
3.withdraw
4.checkbook
5.exit
6.
enter amount to be added:
7.
1.addBalance
2.displayBalance
3.withdraw
4.checkbook
5.exit
6.
enter amount to be withdrawn:
79
Amount: 48922.0
1.addBalance
2.displayBalance
3.withdraw
4.checkbook
5.exit
6.
Enter the name of the receiver:
Enter amount to be debited to receiver:
79
Amount of 79.0 sent to her
Balance: 48847.0
```

WRITEUP:

Expt. No Lab Programs

Date _____

Page No. _____

```
import java.util.Scanner;
abstract class Account {
    String c_name, acc_type;
    int acc_num;
    double balance;
    int minbalance = 2000;
    Account (String c_name, int acc_num,
              double balance)
    { this.c_name = c_name;
      this.acc_num = acc_num;
      this.balance = balance;
      this.acc_type = acc_type;
    }
    abstract void addbal (double amount);
    abstract void display();
    abstract void withdraw (double amount);
}
class curr_acct extends Account {
    curr_acct (String c_name, int acc_num,
               double balance)
    { super(c_name, acc_num, balance);
      System.out.println ("Details of the customer");
      System.out.println ("Customer name" +
                          c_name + " Account number" +
                          acc_num +
                          "Balance" + balance + "Account type: current");
    }
}
```

Teacher's Signature _____

```
void addbal (double amount) {  
    this.balance += amount;  
}
```

```
void display () {  
    System.out.println ("The balance is "+this  
        .balance);  
}
```

```
void withdraw (double amount) {  
    if (this.balance < minbalance) {  
        this.balance = this.balance - this.balance * 0.05;  
        System.out.println ("Updated Balance "+this.balance);  
        System.out.println ("Cannot withdraw");  
    }  
}
```

```
else if (balance > minbalance) {  
    this.balance = this.balance - amount;  
    System.out.println ("Balance is "+this.b  
    alance);  
}
```

```
void interest (double amount) {  
    int time = 3, n = 3;  
    System.out.println ("Rate of interest 0.02");  
    this.balance = this.balance * Math.pow  
        ((1 + (0.02) / n), (n * time));  
}
```

class Main

public static void main(String args[]) {

int ch, n = 1;

double amount;

Scanner s1 = new Scanner(System.in);

curr_acct c = new curr_acct("Adh, 1234567890");

sav_acct s = new sav_acct("Adh, 1234567890");

System.out.println("Press 1 for Current Acc");

Balance press for Saving account");

ch = s1.nextInt();

switch(ch) {

Case 1: System.out.println(" for Current Acc");

while(n != 0) {

c10 = s1.nextInt();

String receiver;

double recamount;

switch(c10) {

Case 1: amount = s1.nextDouble();

c.addBal(amount);

break;

Case 2:

c.display();

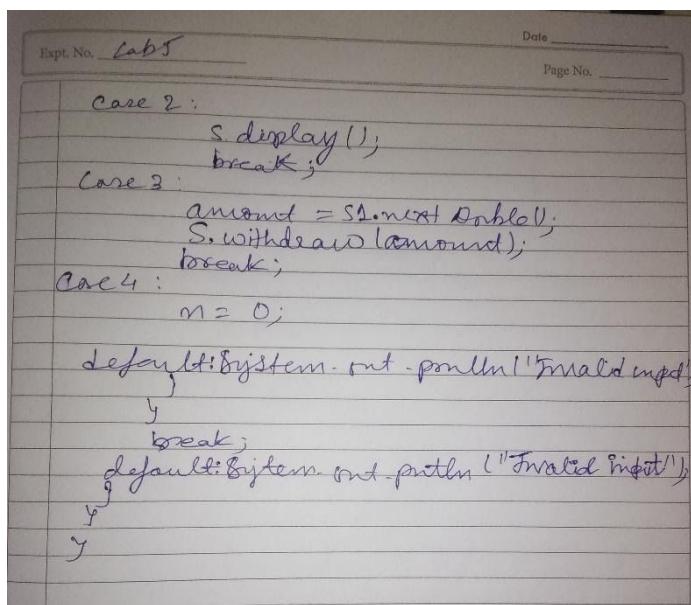
break;

Case 3:

amount = s1.nextDouble();

c.withdraw(amount);

break;



LAB 6

Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class Internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

Student.j
ava

```
package CIE;

import java.util.Scanner;

public class Student
{ public String usn, name;
  public int sem;
}
```

```
*****
*****
Internals.java
```

```

package CIE;

import java.util.Scanner;

public class Internals extends Student
{ public int cie[] = new int[5];
  Scanner get = new Scanner(System.in);

  public void geti()
  { System.out.println("Enter Details: ");
    System.out.println("USN :");
    usn = get.nextInt();
    System.out.println("NAME :");
    name = get.next();
    System.out.println("SEMESTER :");
    sem = get.nextInt();
    System.out.println("CIE MARKS :");
    for(int i=0;i<5;i++)
    { System.out.println("Subject "+(1+i));
      cie[i] = get.nextInt(); }
  }
  public void dispi()
  { System.out.println("\nUSN :" + usn);
    System.out.println("NAME :" + name);
    System.out.println("SEMESTER :" + sem);
    System.out.println("CIE :");
    for(int i=0;i<5;i++)
    { System.out.println(cie[i]); }
    System.out.println("SEE :");
  }
}

```

```

*****
*****
```

Externals.java

```

package SEE;
import CIE.*;
import java.util.Scanner;
```

```

public class Externals extends CIE.Student
{ public int see[] = new int[5];
  Scanner get = new Scanner(System.in);

  public void getm()
  { for(int i=0;i<5;i++)
    { System.out.println("Subject "+(1+i));
      see[i]=get.nextInt(); }
  }
  public void dispsm()
  {for(int i=0;i<5;i++)
   { System.out.println(see[i]); }

}

*****
*****
```

marks.java

```

import java.util.Scanner;
import CIE.*;
import SEE.*;

class marks
{ public static void main(String args[])
  { Scanner get = new Scanner(System.in);
    int n;
    System.out.println("Enetr the no of students: ");
    n = get.nextInt();
    CIE.Internals ints[] = new CIE.Internals[n];
    SEE.Externals exts[] = new SEE.Externals[n];
    for(int i=0;i<n;i++)
      { ints[i] = new CIE.Internals();
        exts[i] = new SEE.Externals();
        ints[i].geti();
        System.out.println("SEE MARKS :");
        exts[i].getm();
      }
    for(int i=0;i<n;i++)
    { ints[i].dispi();
      int total=0;
      exts[i].dispdm();
      for(int j=0;j<5;j++)
        { total=total+ints[i].cie[j]+(exts[i].see[j]/2); }
      System.out.println("TOTAL MARKS : "+total);
    }
  }
```

}

OUTPUT

```
C:\Users\Aditi\Desktop>cd labn
C:\Users\Aditi\Desktop\labn>javac marks.java
C:\Users\Aditi\Desktop\labn>java marks
Enter the no of students:
1
Enter Details:
JSSN :
999
NAME :
ari
SEMESTER :
5
CIE MARKS :
Subject 1
44
Subject 2
33
Subject 3
50
Subject 4
21
Subject 5
44
SEE MARKS :
Subject 1
78
Subject 2
88
Subject 3
88
Subject 4
77
Subject 5
78

JSSN :999
NAME :ari
SEMESTER :5
CIE :
44
33
50
21
44
SEE :
78
88
88
77
78
TOTAL MARKS : 83
TOTAL MARKS : 77
TOTAL MARKS : 99
TOTAL MARKS : 59
TOTAL MARKS : 83
C:\Users\Aditi\Desktop\labn>
```

WRITEUP:

Date _____
Page No. _____

Lab program 6

student.java

```
package CIE;  
import java.util.Scanner;  
public class student {  
    public String usn, name;  
    public int sem;
```

— x — x — x — x —

internals.java

```
package CIT;  
import java.util.Scanner;  
public class Internals extends student {  
    public int cie[] = new int[5];  
    Scanner get = new Scanner(System.in);  
  
    public void geti()  
    {  
        System.out.println("Enter details");  
        System.out.println("USN");  
        usn = get.next();  
        System.out.println("NAME");  
        name = get.next();  
        System.out.println("SEM");  
        sem = get.next();  
    }
```

Teacher's Signature _____

Date _____
Page No. _____

```
system.out.println ("CIE MARKS");
for (int i = 0; i < 5; i++)
{
```

```
    system.out.println (cie [i]);
    system.out.println ("SEE:");
}
```

Externals.java

```
package SEE;
import CIE.*;
import java.util.Scanner;
public class Externals extends CIE.Student
{ public int see [] = new int [5];
  Scanner get = new Scanner (System.in);
```

```
  public void getm ()
  { for (int i = 0; i < 5; i++)
  {
    System.out.println ("Subject" +(i+1));
    see [i] = get.nextInt ();
  }
```

```
  public void disp ()
  { for (int i = 0; i < 5; i++)
  {
    System.out.println (see [i]);
  }
```

Teacher's Signature _____

Date _____
Page No. _____

marks.java

```

import java.util.Scanner;
import CIE.*;
import SEE.*;

class marks
{
    public static void main(String args[])
    {
        Scanner get = new Scanner(System.in);
        int n;
        System.out.println("Enter the students");
        n = get.nextInt();
        CIEInternals ints[] = new CIEInternals[n];
        ents[i] = new
        SEEExternals ents[] = new SEEExternals[n];
        for(int i = 0; i < n; i++)
        {
            ints[i] = new CIEInternals();
            ents[i] = new SEEExternals();
            ints[i].get();
            ents[i].getm();
        }
        System.out.println("SEE marks");
        ents[i].getm();
    }

    for(int i = 0; i < n; i++)
    {
        ints[i].display();
        int total = 0;
        ents[i].display();
        for(int j = 0; j < 5; j++)
        {
            total = total + ints[i].cie[j] + ents[i].see[j];
        }
        System.out.println("Totalmarks" + total);
    }
}

```

Teacher's Signature _____

LAB 7

Write a program to demonstrate generics with multiple object parameters.

```

class
Generics<T,
U,S,B,D>{
    T o1;

```

```
U o2;
S o3;
B o4;
D o5;
Generics(T o1, U o2,S o3,B o4,D o5){
    this.o1 = o1;
    this.o2 = o2;
    this.o3 = o3;
    this.o4 = o4;
    this.o5 = o5;
}
public void print(){
    System.out.println(o1);
    System.out.println(o2);
    System.out.println(o3);
    System.out.println(o4);
    System.out.println(o5);
}
}

public class Gmain{
    public static void main (String[] args){
        Generics <String, Integer, String, Boolean, Double> o =
            new Generics<String, Integer, String, Boolean,
Double>("Hello", 9,"World",true,6.45);

        o.print();
    }
}
```

OUTPUT :

```
Command Prompt
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Aditi>cd
C:\Users\Aditi

C:\Users\Aditi>cd desktop

C:\Users\Aditi\Desktop>javac Gmain.java

C:\Users\Aditi\Desktop>java Gmain
Hello
9
World
true
6.45

C:\Users\Aditi\Desktop>
```



WRITEUP:

Lab prog 7

Date _____

Page No. _____

class Generics<T, U, S>

{
 T o1;
 U o2;
 S o3;

 Generics (T o1, U o2, S o3){

 this . o1 = o1;

 this . o2 = o2;

 this . o3 = o3; } }

 public void print() {

 System . out . println (o1);

 System . out . println (o2);

 System . out . println (o3);

} }

public class Main{

 public static void main (String [] args)

 Generics < String, Integer, String > o =

 new Generics < String, Integer, String >

 ("Hello", 9, "World");

 o . print ();

} }

Teacher's Signature _____

LAB 8

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called "Father" and derived class called "Son" which extends the base class. In Father class, implement a constructor which takes the age and throws the exception Wrong Age() when the input age=father's age.

```
Class
Father

{

    static void acceptNameF(int inputAge) throws ArithmeticException
    {
        try
        {
            if(inputAge<0)
                throw new ArithmeticException("Wrong Age");
        }
        catch (ArithmeticException e) {
            System.out.println("Caught " + e);
        }

    }

}

class Son extends Father
{



    static void CheckSFage(int S_Age, int F_Age) throws
ArithmeticException


    {
        try{
            if(S_Age>=F_Age)
                throw new ArithmeticException("Son's age should be smaller than
father's age ,wrong age");
            System.out.println("Son's age is"+S_Age+"Fathers age is "+F_Age);

        }

        catch (ArithmeticException e) {
            System.out.println("FCaught " + e);
        }

    }

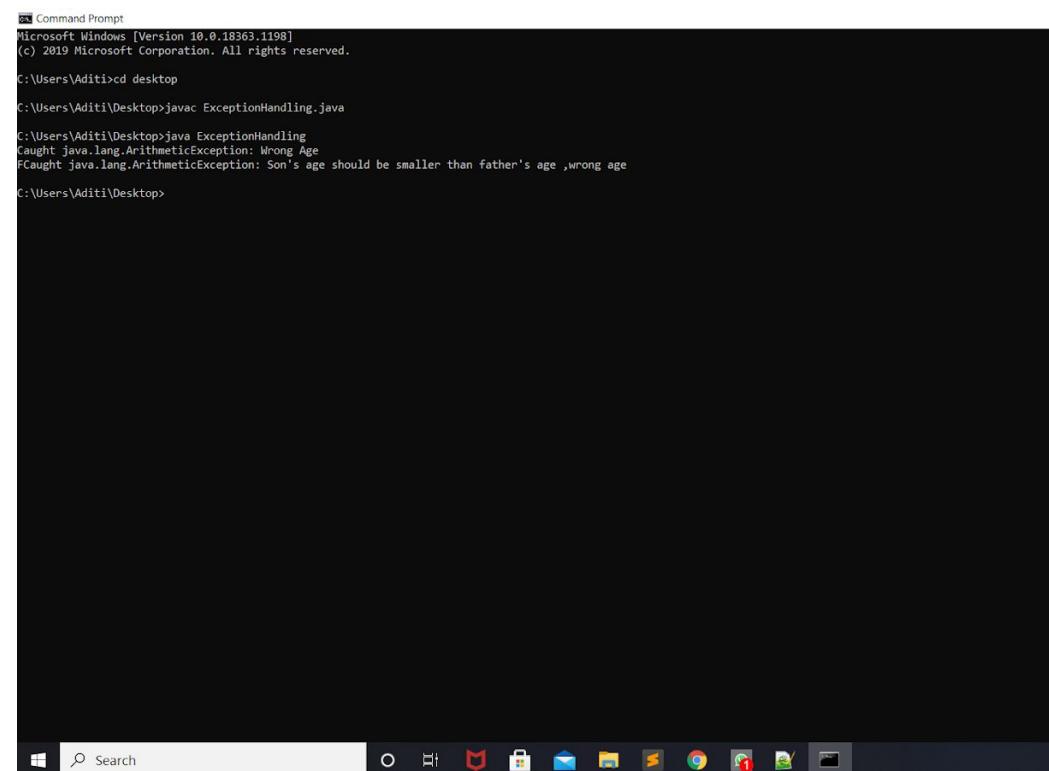
}

public class ExceptionHandling{
    public static void main(String args[]) {
```

```
Father.acceptNameF(-1);
Son.CheckSFage(40,20);

}
}
```

OUTPUT:



```
Command Prompt
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Aditi>cd desktop
C:\Users\Aditi\Desktop>javac ExceptionHandling.java
C:\Users\Aditi\Desktop>java ExceptionHandling
Caught java.lang.ArithmetricException: Wrong Age
Caught java.lang.ArithmetricException: Son's age should be smaller than father's age ,wrong age
C:\Users\Aditi\Desktop>
```

Date _____
Page No. _____

Lab program 8

```

class Father
{
    static void acceptNameF(int inputAge)
    {
        throws ArithmeticException
    }

    try
    {
        if (inputAge < 0)
            throws new ArithmeticException ("Wrong Age");
    }

    catch (Arithmeticeception e)
    {
        System.out.println ("Caught" + e);
    }
}

class Son extends Father
{
    static void CheckSFage(int S_Age, int F_Age)
    {
        throws ArithmeticException
    }

    try
    {
        if (S_Age > F_Age)
            throws new ArithmeticException ("Son's age should be smaller than father's age, wrong age");
    }

    catch (Arithmeticeception e)
    {
        System.out.println ("Son's age is " + S_Age + " Father's age " + F_Age);
    }
}

public class Exception Handling
{
    public static void main (String args[])
    {
        Father.acceptNameF (-1);
        Son.CheckSFage (40, 20);
    }
}

```

Teacher's Signature _____

LAB9

Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

```

class
Thread1
implement
S
Runnable

{

    String name;
    Thread t;
    int time;
}

```

```

        Thread1(String threadname,int time)
        {
            name = threadname;
            this.time=time;
            t = new Thread(this, name);
            t.start();
        }

    public void run()
    {
        try {
            for(int i =100; i > 0; i--)
            {
                System.out.println(name);
                Thread.sleep(time);
            }
        }
        catch (InterruptedException e) {
            System.out.println(name + "Interrupted");
        }
        System.out.println(name + " exiting.");
    }
}

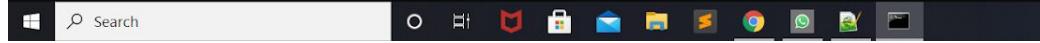
class threadmain{
public static void main(String args[])
{
    Thread1 t1=new Thread1("BMS COLLEGE OF
ENGINEERING",10000);
    Thread1 t2=new Thread1("CSE",2000);
}
}

```

OUTPUT:

```
Command Prompt - java threadmain
Microsoft Windows [Version 10.0.18363.1198]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\Aditi\cd desktop
C:\Users\Aditi\Desktop>javac threadmain.java
C:\Users\Aditi\Desktop>java threadmain
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
BMS COLLEGE OF ENGINEERING
CSE
CSE
CSE
CSE
CSE
```



WRITEUP:

lab program 9

Date _____

Page No. _____

class Thread1 implements Runnable

{

String name;

Thread t;

int time;

Thread1(String threadname, int time)

{

name = threadname;

this.time = time;

t = new Thread(this, name);

t.start();

}

public void run()

{

try { for (int i = 100; i > 0; i--)

System.out.println(name);

Thread.sleep(time);

} catch (InterruptedException e) {

System.out.println(name + " Interrupted");

} system.out.println(name + " exiting");

}

class Threadmain

public static void main (String args[])

{ Thread t1 = new Thread1("BMS", 10000);

Thread t2 = new Thread2("CSE", 2000);

}

}

Teacher's Signature _____

LAB 10

Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an ArithmeticException. Display the exception in a message dialog box.

```
import  
java.aw  
t.*;  
import java.awt.event.*;  
public class Divisions extends Frame implements ActionListener  
{  
    Dialog d;  
    TextField Num1,Num2,result;  
    Button Divide;  
    public Divisions()  
    {  
        setLayout(new FlowLayout());  
        setSize(500,500);  
        Num1=new TextField(10);  
        Num2=new TextField(10);  
        result=new TextField(10);  
        Divide=new Button("DIVIDE ");  
        add(new Label("Enter 1st number : "));  
        add(Num1);  
        add(new Label("Enter 2nd number: "));  
        add(Num2);  
        add(new Label("Result : "));  
        add(result);  
        add(Divide);  
        Divide.addActionListener(this);  
        setVisible(true);  
        addWindowListener(new MyWindowAdapter());  
    }  
    public void actionPerformed(ActionEvent ae)  
    {
```

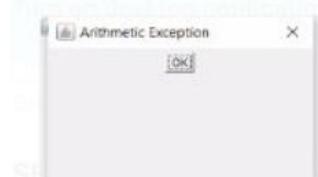
```

if(ae.getSource()==Divide)
{ try
{
result.setText(Integer.toString((Integer.parseInt(Num1.getText().trim()))/(Integer.parseInt(Num2.getText().trim()))));
}
catch(ArithmaticException aex)
{ Dia d1=new Dia("Arithmatic Exception");
d1.setVisible(true);
}
catch(NumberFormatException nfe)
{
Dia d2=new Dia("Number Format Exception ");
d2.setVisible(true);
}
}
public static void main(String args[])
{
new Divisions();
}
}
class Dia extends Dialog implements ActionListener
{
Button ok;
Dia(String str)
{
super(new Frame(),str,true);
ok=new Button("OK");
setLayout(new FlowLayout());
setSize(300,200);
add(ok);
ok.addActionListener(this);
addWindowListener(new MyWindowAdapter());
}
public void actionPerformed(ActionEvent ae)
{
setVisible(true);
}
class MyWindowAdapter extends WindowAdapter{
public void windowClosing(WindowEvent we)
{
System.exit(0);
}
}
}

```

OUTPUT:

com.vitoshkaa.java.exceptions



Enter 1st number: Enter 2nd number: Result: DIVIDE

SE

M

SE

PR

1

SE

SE</

WRITEUP:

Date _____
pt. No. Lab Book 10 Page No. _____

```
import java.awt.*;
import java.awt.event.*;
public class Divisions extends Frame
    implements ActionListener
{
    Dialog d;
    JTextField Num1, Num2, result;
    JButton Divide;
    public Divisions()
    {
        setLayout(new FlowLayout());
        setSize(300, 300);
        Num1 = new JTextField(10);
        Num2 = new JTextField(10);
        result = new JTextField(10);
        Divide = new JButton("DIVIDE");
        add(new Label("Enter 1st no."));
        add(Num1);
        add(new Label("Enter 2nd number"));
        add(Num2);
        add(result);
        add(Divide);
        Divide.addActionListener(this);
        setVisible(true);
        addWindowListener(new MyWindowAdapter()
            (new MyWindowAdapter()));
    }
}
```

```

public void actionPerformed(ActionEvent ae)
{
    if (ae.getSource() == Divide)
    {
        result.setText(Integer.parseInt(
            (Num1.getText().trim()) / Integer.parseInt(
                Num2.getText().trim())));
    }
}

```

Catch Arithmetic Expression

Catch(ArithmeticException e)

```

Dia d1 = new Dia("Arithmetic Exception");
d1.setVisible(true);

```

Catch(NumberFormatException e)

```

Dia d2 = new Dia("Number Format Exception");
d2.setVisible(true);
}

```

public static void main(String args[])

```

new Division();
}

```

class Dia extends Dialog implements

ActionListener {

ButtonOK;

Dia(str);

```

super(new Frame(), str, true);

```

OK = new Button("OK");

setLayout(new GridLayout());

setSize(300, 200);

add(OK);

OK. add ActionListener(this);
add WindowListener(new MyWindowAdapter());
public void actionPerformed(ActionEvent e)
{
 setVisible(false);
}
class MyWindowAdapter extends WindowAdapter
{
 public void windowClosing(WindowEvent we)
 {
 System.exit(0);
 }
}
